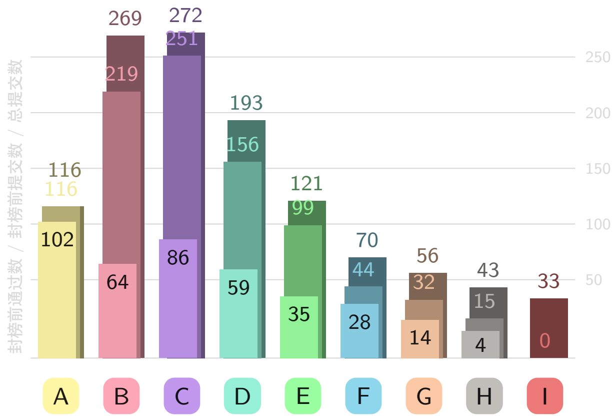
第十九届“快手杯”
北京航空航天大学程序设计竞赛
决赛题目解析

2024 年 12 月 15 日

统计



封榜前通过数 / 封榜前提交数 / 总提交数
250
200
150
100
50
116
116
102
269
219
64
272
251
86
193
156
59
121
99
35
70
44
28
56
32
14
43
15
4
33
0
A
B
C
D
E
F
G
H
I

最强辅助

对于一个 n 位数 c 划分为一个 n_1 位数 a 和一个 n_2 位数 b 的情形，有 $n = n_1 + n_2$ 且 $c = a \times 10^{n_2} + b$。考虑划分前后的伤害：划分前为 c，划分后为 $a \times b$。由于 b 是 n_2 位数，$b < 10^{n_2}$，故

$$c = a \times 10^{n_2} + b > a \times 10^{n_2} > ab$$

因此不划分总是最优的。只需要读入 S 然后输出 S 即可。注意 S 的范围，需要使用字符串或者对应语言的 64 位整数类型。

四舍五入

容易证明，大于 $\underbrace{4\cdots4}_{y-1}5\underbrace{0\cdots0}_{x-y}$ 的数均满足条件，等差数列求和即可。

不够聪明的贝贝

一道比较清新的构造题。

- 当 n 为偶数时，令所有 $a_i = -1$，此时必然有 $A = (-1)^n = 1, B = (-1)^{n+1} = -1$，显然有 $1 > -1$；
- 当 n 为奇数时，令所有 $1 \le i < 2n+1$ 的 a_i 都有 $a_i = -1$，且 $a_{2n+1} = 2$。
  - 当贝贝选择到了 a_{2n+1}，那么 $A = 2 \times (-1)^{n-1} = 2, B = (-1)^{n+1} = -1$，显然有 $2 > -1$；
  - 当贝贝没有选择到 a_{2n+1}，于是 $A = (-1)^n = -1, B = 2 \times (-1)^n = -2$，显然有 $-1 > -2$；

不够聪明的贝贝

类似地，可以拓展出通解：

- 对于正整数 $d \le 9$：
  - $a_i = -d,\ 1 \le i < 2n+1$；
  - a_{2n+1} 选取一个大于 d^2 的数字即可。

阿斯特赖亚

可以使用二分答案来转化为判定性问题。二分强化的次数 T，判断是否可以完成 T 次强化。

在判定问题中，为了完成 T 次强化，每个种类的货币需要的数量是已知的，第 i 种货币需要的数量是 $\sum_{j=1}^{T} c_j[t_j = i]$。求出需要的货币数量之后，会有一些货币不够，而另一些货币有多余。此时把多余的货币全部换成缺少的货币再进行强化是最优的策略。统计每种货币的多余部分除以 3 之和以及缺少部分之和，判断多余部分能否兑换填上缺少部分即可。

该算法的时间复杂度为 $O((n + K)\log n)$，可以通过本题。

阿斯特赖亚

更快的做法是，考虑二分时统计的东西，每种货币的多余部分除以 3 之和以及缺少部分之和，两个数都可以在从 1 到 n 扫描的时候实时维护。每次修改一种货币的数量时，更改它对这两个数的贡献即可，可以做到实时判断。

改进后时间复杂度为 $O(n)$。

工作调度

如果没有工作类型的限制，用拓扑排序算法即可解决。如果有，那么问题的关键就是决定工作类型的先后顺序。为此，我们可以对工作类型建图。我们创建一个新图，令每个点代表一种工作类型，如果工作 y_i 依赖工作 x_i，且两个工作的类型不相同，那么建一条边 $a_{x_i} \to a_{y_i}$，表示必须先完成类型为 a_{x_i} 的工作，才能完成类型为 a_{y_i} 的工作。通过在这个图上应用拓扑排序，得到的拓扑序即为一种合法的工作类型完成顺序。

工作调度

有了这个顺序之后，为了得到一种合法的工作安排方案，我们还需要对拓扑排序算法略作修改。具体来说，在得到了工作类型拓扑序后，依次考虑每个工作种类，每次仅对该种类的节点进行拓扑排序，即，在将每个节点入队列时，除了要求其入度为零外，还需要其工作种类与当前考虑的工作种类相符。

时间复杂度为 $O(n+m)$。

数数

对于“至少有一段”的情况计数，不难想到将其转化求“没有任何一段”的数量。

对相反的情况进行 DP，令 $f_{i,j}$ 表示第 i 位置填了 j，且前缀 $[1,i]$ 中不包含一段连续的长度 $\geq a$ 的 ‘0’ 且不包含连续长度 $\geq b$ 的 ‘1’ 的方案数。

转移的时候用所有情况减去新出现一个合法段的情况。比如求 $f_{i,0}$，从 $i-1$ 转移过来的总数为 $f_{i-1,0}+f_{i-1,1}$；在 $[i-a+1,i]$ 区间可能出现一段长度为 a 的 ‘0’，总数为 $f_{i-a,1}$。因此得到 $f_{i,0}=f_{i-1,0}+f_{i-1,1}-f_{i-a,1}$。$f_{i,1}$ 的转移类似。

总复杂度 $O(n)$。

回文串

根据题意，一个好的字符串中只含有一个本原回文子串，也就是只含有一个形如 aa 或 aba 的回文中心。我们可以按照顺序枚举 s_i 作为可行子串的右端点。记录 p_1 表示最近的回文中心的左端点，即 aa 或 aba 中第一个 a 所在的位置，p_2 为次近的回文中心的左端点。此时，对于该右端点，区间 $[p_2, p_1)$ 即为合法左端点的集合。由此计数即可，复杂度 $O(|S|)$。

急死地

通过数字划分的状态去判定 $\gcd > 1$ 并不现实。因此考虑先定 gcd，再确定数字的划分。

枚举 a_1 所在集合 A 的 gcd，对每一个 a_1 的约数 d_i 进行处理：判断每个数是否能被划分到 A，其余的都必须划分到 B，设其余数的 gcd 为 g，划分合法的前提是 $g > 1$。但还有一些数可以划分到 A 也可以划分到 B，于是再枚举 g 的因子作为 B 的 gcd，看看哪些数可以从 A 跑到 B 里，尽量让 A, B 的大小平均。设约数个数最多为 D，上述做法复杂度 $O(D^2 n)$。

实际上，并不需要枚举所有因子。仅枚举质因子就能包含最优的情况。复杂度 $O(\omega^2 n)$，ω 为质因子的数量，最大为 10。

倒反天罡

令 $d=r-l+1$，首先考虑 $k=d$ 的情形。最优的年龄阈值可以通过枚举求出。注意到枚举年龄阈值 a 时，a 变化到 $a+1$ 时仅有 $age=a+1$ 的猫影响所关注的「倒反天罡值」，并且每只猫的贡献为 ± 1 之一的固定值。因此可以针对值域维护贡献值，查询前缀（或者后缀）最值即可回答询问。利用莫队离线处理询问，时间复杂度 $O(q\sqrt{n}\log n)$。

对于 $k \neq d$ 的情形，设 $k=d-c$ 并且设相同区间 $k'=d$ 的答案为 ans，容易证明所求答案为 $\max(0, ans-c)$。